Pinball

Software Requirments Specification

Version 1.0

21-10-2015

Group 21
The Pirates

Prepared for
CS 251 – Software Systems Lab
Instructor: Sharath Chandran
Autumn 2015

Contents

1 Introduction 1

1.1 Purpose 1

1.2 Scope 1

1.3 Definitions, Acronyms, and Abbreviations 1

1.4 References 3

1.5 Overview 3

2 General Description 3

2.1 Product Perspective 3

2.2 Product Functions 3

2.3 User Characteristics 3

2.4 Assumptions and Dependencies 4

3 Specific Requirements 4

3.1 External Interface Requirements 4

3.1.1 User Interfaces 4

3.1.2 Hardware Interfaces 4

3.1.3 Software Interfaces 4

3.2 Functional Requirements 4

4 Analysis Models 5

4.1 Call Graph 5

5 Revision History 5

6 Document Approval 5

7 Deviations from ideal 6

8 Difficulties 6

9 Honor Code 7

1 Introduction

The purpose of this document is to present a detailed description of the Web Publishing System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended to contain all of the information needed by a software engineer to adequately design and implement the game of pinball.

1.1 Purpose

The game is designed as a part of the first project involving box2d for CS251.

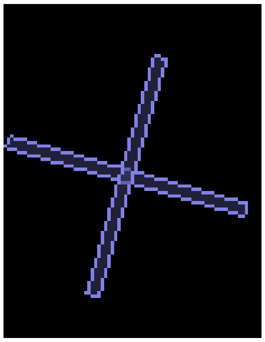
1.2 Scope

The scope of the project is to turn the age old implementation of the arcade pinball game into a computer based game using box2d physics engine. The game is made cross-platform by the use of cmake.

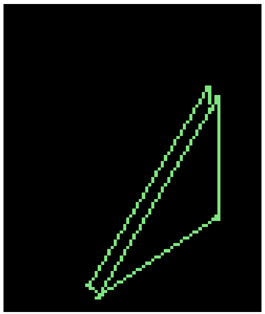
1.3 Definitions, Acronyms, and Abbreviations

Flipper Wheel

Rotating wheel hinged about it's center.



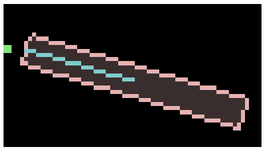
Sling Shot



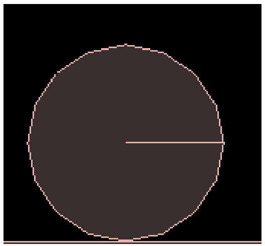
Elastic triangular objects.

Flipper Bats

A tapered object that forms the primary means of hitting the ball



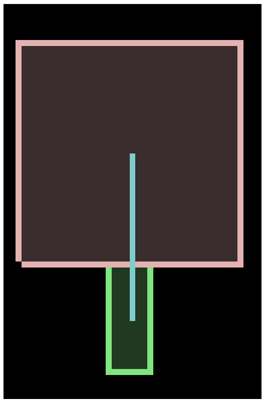
Ball



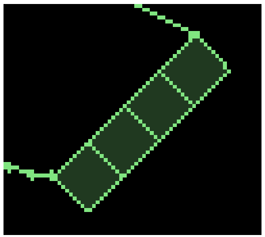
The circular object that moves around the board

Launcher

Provides the initial momentum to the ball propelling it upward.



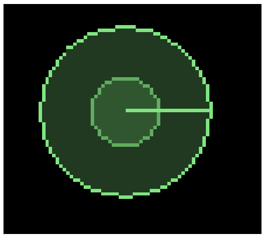
Bumpers



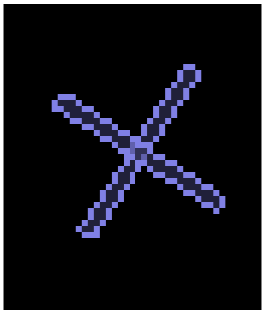
Rectangular reflectors from which the ball bounces of.

Shooting Bumpers

Circular objects from which the balls bounce off.



Rotators



Continuously rotate in either one of the directions.

1.4 References

[?] [?] [?] [?] [?]

1.5 Overview

In this report we have given a gist of our design, its salient features and working parts. The next sections consists of a general discussion about our project as well as the product functions.

2 General Description

2.1 Product Perspective

Pinball was first designed as an arcade game in the 19th century. With the advent of the computers the game was also implemented as a computer game. We intend to develop our own version of the game.

2.2 Product Functions

The major game functions are:

Launching the ball	The ball is launched on user giving the command.
Flipper movement	The corresponding flippers are moved based on user input.
Rotate wheels	The direction of rotation of the wheels is changed based on user input.
Restart	The game is to be restarted whenever the user deems right by taking in input.
Pause	Pause the game whenever commanded to.

2.3 User Characteristics

The user is expected to know the controls of the game, how to operate the keyboard and Ubuntu literate in the sense that he knows how to run the game from the terminal using the bash script provided (or otherwise).

2.4 Assumptions and Dependencies

A) Asssumptions

We assume that the user will restart the game whenever he deems right.

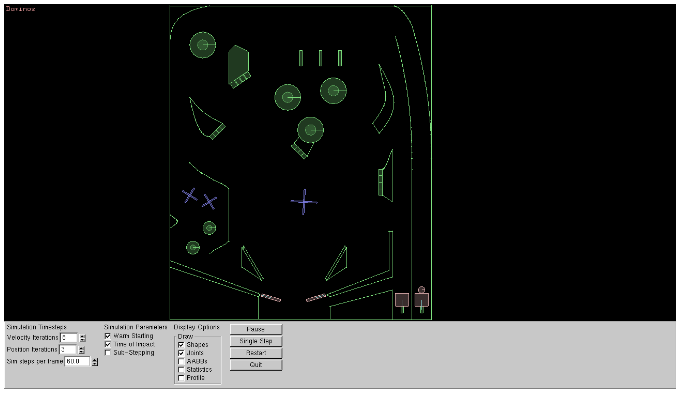
B) Dependencies

The speed of execution depends on the processor speed in which the game is being played.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces



Simulation Timesteps
Velocity Iterations 8
Position Iterations 3
Sim steps per frame 60.0

Simulation Parameters
☑ Warm Starting
☑ Time of Impact
□ Sub-Stepping

Display Options
Draw
☑ Shapes
☑ Joints
□ AABBs
□ Statistics
□ Profile

3.1.2 Hardware Interfaces

A working keyboard is sufficient to interact with the game.

3.1.3 Software Interfaces

cmake is essential to run the game. A C++ compiler is also necessary to create an executable for the game.

3.2 Functional Requirements

On button press:

W :	The launcher on the right is activated and gives a push to the ball.
A :	The lift flipper turns anti clockwise.
D :	The right flipper turns clockwise.
R :	The game is restarted from the start.
P :	The game is frozen in its current state.
esc :	Quit game console.
8 :	The launcher on the left is activated.
4 :	Change the direction of flipper wheel to counter clockwise.
6 :	Change the direction of flipper wheel to clockwise.

4 Analysis Models

4.1 Call Graph

5 Revision History

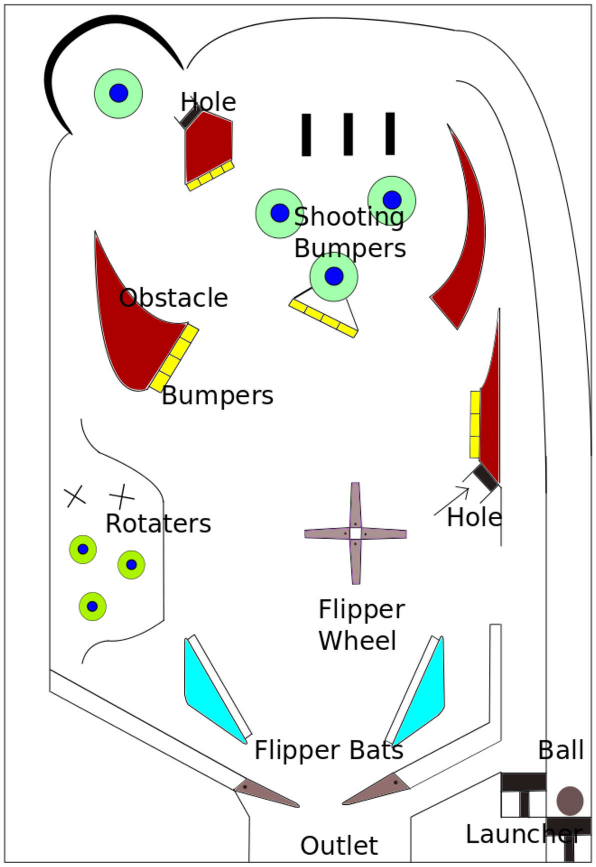
Date	Description	Author	Comments
17-10-2015	Draft of SRS	Sai Teja	Revision Done
19-10-2015	Added images	Chanukya	Revision Done
20-10-2015	Webpage	Naveen	Revision Done

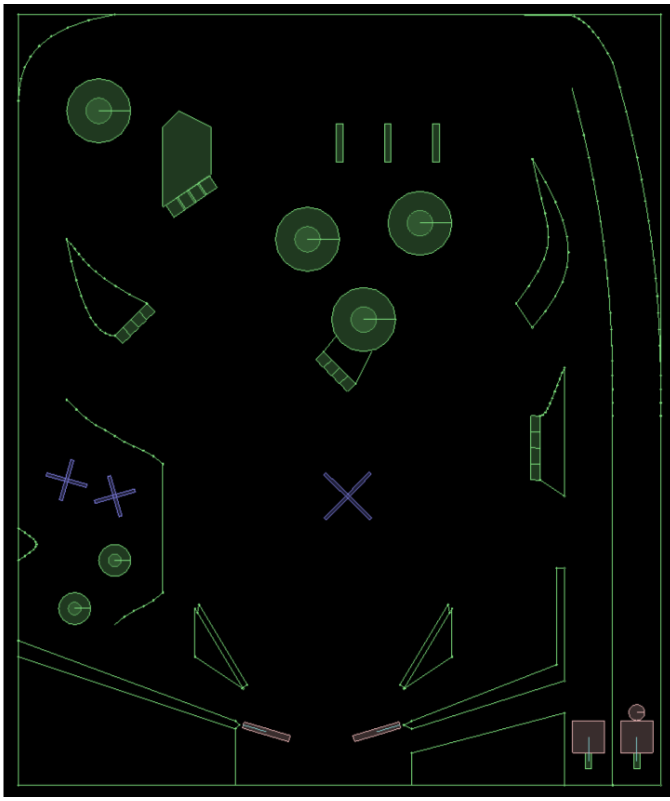
6 Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
	Prof. Sharat Chandran	Instructor, CS 251	

7 Deviations from ideal





The first figure shows the initial design that we thought of creating for our project.
The second figure shows the final design that we created.
We were not able to create a hole as we were not able to detect collisions.

8 Difficulties

We faced some difficulties in giving the keyboard functions to the flipper bats and launcher.
We overcame this problem by using an extern variable so that it can be used over different files and at desired locations.

9 Honor Code

We pledge on our honor that we have not given or received any unauthorized assistance on this project.